METHOD AND SYSTEM FOR CREATING AN AUTOMATICALLY ADJUSTING USB MASS STORAGE DEVICE

BACKGROUND OF THE INVENTION

1. Technical Field:

[0001] This invention relates generally to network computing systems, and in particular to remotely managed computers. Still more particularly, the present invention relates to a method and system for dynamically configuring a Universal Serial Bus (USB) device interface to flexibly make available, to a remotely managed computer, mass storage devices that are mounted on a server.

2. Description of the Related Art:

[0002] While early computers were "stand alone" and unable to communicate with other computers, most computers today are able to communicate with other computers for a variety of purposes, including sharing data, e-mailing, downloading programs, coordinating operations, etc. Each computer typically connects to a network via a Network Interface Card (NIC) or an integrated Local Area Network (LAN) On Motherboard (LOM). The network couples multiple computers and/or peripheral devices (such as printers, secondary storage, etc.) The network may be a Local Area Network (LAN) or a Wide Area Network (WAN).

[0003] A LAN is a small local network, wired or wireless, that is typically used by a single enterprise. Such LANs, using architectures such, as an Ethernet, Token Ring, ARCnet or a wireless protocol such as 802.11a/b/g, are used to connect computers and peripheral devices within a limited area, such as a single campus or building.

[0004] Computers are not limited to the confines of a LAN. Rather, a computer can be connected to another computer or peripheral device via a wide area network (WAN), such as the Internet, to provide a communications link over a larger area. Each computer can be directly connected to a WAN via a Network Interface Card (NIC) or LOM, or the computer can be

indirectly connected to the WAN via the LAN. The WAN can be wired or wireless, similar to that described for the LAN.

[0005] One advantage of a network based computer system is that resources, including peripherals such as hard drives, printers, etc., can be shared on the network. A typical way of sharing such resources is to assign an Internet Protocol (IP) address to each peripheral, and authorize certain computers on the network to access that peripheral. However, such a system requires extensive command and control resources to avoid corruption and cross talk of the resource, especially if the resource is a storage device.

[0006] What is needed, therefore, is a method and system for a remote computer to utilize a remote mass storage device without the need for cumbersome IP addresses, etc. Preferably, such a system would be dynamically flexible, so that different numbers of drives can be emulated by the remote computer according to its needs.

SUMMARY OF THE INVENTION

[0007] As will be seen, the foregoing invention satisfies the foregoing needs and accomplishes additional objectives. Briefly described, the present invention provides a method and system for virtually mounting Mass Storage Devices (MSDs) that correspond to physically mounted MSDs that are remotely mounted on an administrative computer.

[0008] A user mounts multiple drives on an administrative computer, according to how many drives will be needed by a client bootable computer. A signal is sent to a Universal Serial Bus (USB) Mass Storage Device Interface (MSDI) on the client computer, indicating how many MSDs are mounted and available on the administrative computer. The USB MSDI signals a USB Emulated Storage Device (ESD) on the client computer to disconnect from the client computer while the USB MSDI reconfigures to show the appropriately set maximum Logic Unit Number (LUN) corresponding with the multiple MSDs mounted on the administrative computer. The USB ESD then reconnects as a new USB device that has morphed appropriately to show the correct MSDs. The Operating System (OS) or Basic Input/Output System (BIOS) on the client computer sees the new USB device and adds the MSDs to a list of drives connected to the system. The MSDs mounted on the administrative computer are now used by the client computer as if the MDSs were connected directly to the client computer.

BRIEF DESCRIPTION OF THE DRAWINGS

[0009] The novel features believed characteristic of the invention are set forth in the appended claims. The invention itself, however, as well as the preferred modes of use, further objects and advantages thereof, will best be understood by reference to the following detailed description of an illustrative embodiment when read in conjunction with the accompanying drawings, wherein:

[0010] Figure 1 depicts a schematic diagram illustrating a computer network within which the present invention may find application;

[0011] Figure 2 depicts an exemplary bootable computer under boot control of an administrative computer;

[0012] Figure 3 illustrates an exemplary administrative server; and

[0013] Figure 4 is a flow-chart of steps taken in a preferred embodiment of the present invention for virtually mounting mass storage devices on the bootable computer.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

[0014] Referring now to the drawing figures, in which like numerals indicate like elements or steps throughout the several views, a preferred embodiment of the present invention will be described. In general, the present invention provides an improved method and system for emulating mass storage devices.

[0015] With reference now to **Figure 1**, there is depicted an exemplary diagram of an administrative computer **102** coupled to a hyper-secure Remote Supervisor Adapter (RSA) management network **104**, which is coupled to a bootable computer **106**. Bootable computer **106** is preferably a network server that, when booting up or re-booting, is under the control of administrative computer **102**. Thus, the logical relationship between administrative computer **102** and bootable computer **106** is not unlike that of a server and client on a network. Additional details of administrative computer **102** and bootable computer **106** are given below.

[0016] With reference now to **Figure 2**, additional detail of administrative computer **102** is given. A Central Processing Unit (CPU) **202** connects via a processor interface bus **204** (also referred to in the art as a "front side bus," "host bus," or "system bus") to a North Bridge **206**. North Bridge **206** is a chip or chipset arbiter logic circuit having a memory controller **207** connected to a system memory **212**. A video controller **228** is coupled to North Bridge **206** and a video display **230** for viewing a graphical user interface of software operations being performed on administrative computer **102** or remote operations, including booting operations, being performed on remote bootable computer **106**. Also connected to North Bridge **206** is a high speed interconnect bus **208**. North Bridge **206** is connected via interconnect bus **208**, which may be a Peripheral Component Interconnect (PCI) bus, to a South Bridge **210**.

[0017] South Bridge **210** is a chip or chipset Input/Output (I/O) arbiter that includes the necessary interface logic to convey signals from interconnect bus **208** to (typically slower) I/O interfaces, including a Super I/O **216**. Super I/O **216** is preferably a chip or chipset including necessary logic and interfaces for a parallel port **218** and a non-USB (Universal Serial Bus) serial port **220**, as are understood in the art of computer architecture. Super I/O **216** may also include

controllers for non-USB devices such as a keyboard controller **222** for a non-USB keyboard and an Enhanced Integrated Device Electronics (EIDE) port **226**, to which is connected an array of Compact Disk – Read Only Memory (CD-ROM) drives **234a-n**. Also connected to Super I/O **216** is a floppy disk controller **224**. Floppy disk controller **224** supports an interface with one or more floppy disk drives **236**, depicted in exemplary form as an array of floppy drives **236a-n**.

[0018] Coupled with South Bridge **210** is a USB host controller **213**, which provides a USB interface from USB compliant devices (not shown) to administrative computer **102**, including CPU **202**. USB compliant devices may be floppy disk drives, CD-ROM drives, keyboards and other peripheral devices that are configured to comply with the "Universal Serial Bus Specification" release 2.0, April 27, 2000 (USB.org), which release or later is herein incorporated by reference in its entirety. USB host controller **213**, which is likewise USB compliant, may be implemented in a combination of hardware, firmware and/or software.

[0019] As known to those skilled in the art of computer peripheral device interfaces, the USB specification was prepared by representatives of Compaq Computer Corporation, Hewlett-Packard Company, Intel Corporation, Lucent Technologies Inc., Microsoft Corporation, NEC Corporation, and Royal Philips Electronics (Philips). Peripheral device interfaces that comply with the specification are referred to as USB interfaces and have been included in many recently developed personal computer systems. Such USB devices are generally referenced as either low-speed devices, capable of transferring data at a rate of 1.5 Megabits per second (Mb/s); or high-speed devices (also called full-speed devices) capable of transferring data at 12Mb/s. Under the USB 2.0 specification, full-speed devices are capable of using 40x multipliers for a transfer rate of 480Mb/s, and such USB devices are typically known as high-speed devices.

[0020] Within a computer system, a USB interface serves to provide well-known plug-n-play capability for computer peripherals such as external Compact Disc - Read Only Memory (CD-ROM) drives, joysticks, magnetic tape and floppy drives, external hard drives, scanners, and printers. Additionally, the USB interface allows an alternate connection for primary system input devices such as keyboards and mice, providing an alternative to the dedicated keyboard

and mouse non- USB ports that many personal computer manufacturers provide. The industry generally refers to the non-USB keyboard and mouse ports as the PS/2 keyboard and PS/2 mouse port, respectively.

[0021] Coupled to USB Host controller **213** is a Remote Supervisor Adapter (RSA) card **214**, preferably an RSA-II or later version card manufactured by International Business Machines of Armonk, New York. RSA card **214** provides a connection to hyper-secure RSA management network **104** using security methodology known to those skilled in the art of network, and particularly RSA network, security. As described in further detail below, RSA card **214** communicates network packets of information in a hyper-secure manner between administrative computer **102** and bootable computer **106**.

[0022] With reference now to **Figure 3**, there is depicted a block diagram of an exemplary bootable computer **106**. A Central Processing Unit (CPU) **302** connects via a processor interface bus **304** (also referred to in the art as a "front side bus," "host bus," or "system bus") to a North Bridge **306**. North Bridge **306** has a memory controller **307** connected to a system memory **312**. Also connected to North Bridge **306** is a high speed interconnect bus **308**. Also connected to North Bridge **306** is a video controller **328**, which drives a video display **330**.

[0023] When bootable computer **106** is being remotely booted up, as described below, the booting process and progress can be visually monitored by an administrator at administrative computer **102**. This remote visual monitoring can be performed using a video feed, either direct or through a video camera, from video display **330** of bootable computer **106** to video display **230** of administrative computer **102**. North Bridge **306** is connected via interconnect bus **308**, which may be a Peripheral Component Interconnect (PCI) bus, to a South Bridge **310**.

[0024] South Bridge **310** includes the necessary interface logic to convey signals from interconnect bus **308** to a Super I/O **316**. Connected to Super I/O **316** is a modulator/demodulator (modem) **326**, which modulates signals for transmission on telephone and telephone like lines (not shown).

[0025] Coupled with South Bridge **310** is a USB host controller **313**, which provides a USB interface from a USB-Storage Device Emulator (USB-SDE) **314**. USB host controller **313** communicates with USB-SDE **314** as though USB-SDE **314** were a physical USB device(s). That is, USB-SDE **314** provides multiple emulated mass storage devices, such as a CD-ROM, a floppy drive, a ZIP drive, etc. USB host controller **313** communicates with end points "in" and "out," which communicate with a USB Mass Storage Device Interface (USB-MSDI) **336**.

[0026] A USB device **324** includes a USB Mass Storage Device Interface (USB-MSDI) **336**, which provides an interfaces that emulates of an array of MSDs **332** that have been mounted on administrative computer **120**. Emulation data (including Logical Unit Numbers (LUN) identifying a particular physical remote MSD as well as data on that physical remote MSD) is received via a Network Interface Card (NIC) **322**, which receives network packets from RSA management network **104**. The packets are in TCP/IP format, and must be "unwrapped" by a TCP/IP to USB converter **318**, before the data is sent to USB-MSDI **336**.

[0027] Similarly, USB signals sent from USB host controller **313** to USB device **324** must first be converted into TCP/IP format before being sent to administrative computer **102**. To accomplish this, the USB signal is passed through a USB to TCP/IP converter **320**, which "wraps" the USB data in a TCP/IP packet for transmission to administrative computer **102**.

[0028] Note that administrative computer **102** has an array of previously mounted MSDs **332**, which may be floppy disk drives **236** or CD-ROM drives **234** shown in **Figure 2**, or any other mass storage device(s), including but not limited to hard drives, Digital Video Disk (DVD) drives, tape drives, etc. Administrative computer **102** has access to remote disk Java applets **334**, which tell bootable computer **106** how many MSDs are available to bootable computer **106**.

[0029] Referring now to **Figure 4**, there is illustrated a flow-chart describing a preferred embodiment of the present invention. After starting at initiator block **402**, an administrative computer user mounts up to 16 MSDs (drives) on the administrative computer (block **404**). These drives are those that will be available to a remote server computer, which will "see" the drives in a local emulated form.

[0030] As shown at block **406**, the administrative computer then invokes a remote disk Java applet, which sends a command to the USB-MSDI a command indicating how many MSDs are mounted and available to the bootable computer. The USB-MSDI then signals the USB Storage Device Emulator to disconnect from the bootable computer while the USB-MSDI is reconfigured with the appropriate number of mass storage devices according to the maximum Logical Unit Number (block **410**). The USB Storage Device Emulator then reconnects (block **412**) to the bootable computer, with the USB Storage Device Emulator now presenting the correct number of emulated MSDs. The operating system of the bootable computer "sees" the newly morphed USB Storage Device Emulator (block **414**), and adds the MSDs to a list showing the drives connected to the bootable computer. Thus, as shown at block **416**, the MSDs mounted on the remote administrative computer are now used by the bootable computer as if the MSDs were mounted directly on the remote bootable computer. The process then ends at terminator **418**.

[0031] Note that the exemplary embodiments shown in **Figures 2 and 3** are provided solely for the purposes of explaining the invention and those skilled in the art will recognize that numerous variations are possible, both in form and function. All such variations are believed to be within the spirit and scope of the present invention.

[0032] The present invention has been described in relation to particular embodiments that are intended in all respects to be illustrative rather than restrictive. Alternative embodiments will become apparent to those skilled in the art to which the present invention pertains without departing from its spirit and scope. Accordingly, the scope of the present invention is defined by the appended claims rather than the foregoing discussion.